# OPTIMIZED CHAMFER MATCHING FOR SNAKE-BASED IMAGE CONTOUR REPRESENTATIONS

*András Hajdu, Athanasios Roubies, Ioannis Pitas*

Artificial Intelligence and Information Analysis Laboratory,
Department of Informatics, Aristotle University of Thessaloniki, Greece
e-mail: pitas@zeus.csd.auth.gr

## ABSTRACT

In this paper we present a novel method on how to take advantage of the snake representation of target objects, when doing chamfer matching for detection/recognition purposes. In this case several time-consuming steps of classic chamfer matching approaches can be simplified. Moreover, we investigate the possibility of involving less pixels from both the target and template object to speed up computations. We introduce an optimization method for such an object reduction, which is valid also in the general application scheme of chamfer matching. Finally, we present our experimental results regarding human body detection.

## 1. INTRODUCTION

Object detection/classification have been challenging problems in the digital image processing. Usually, some features are extracted from the image (e.g. edges), and then some matching procedure is applied to find the occurrences of pre-defined templates. Such a method is the chamfer matching. Here, usually an edge detection step is applied first to obtain a binary image, and then its distance map is calculated [5, 9]. If the search range cannot be restricted, a scanning step is applied for the whole image to match pre-defined templates, like in [5, 6]. In our currently developed system, we analyze thermal videos captured in a fire scene. The user makes a rough manual selection, and the system extracts the boundary of the nearby object, and tries to classify it. In this paper, we present a method for classifying the object represented by the snake provided by the GVF (Gradient Vector Flow) iteration process, which is noise tolerant and is able to travel large distances from the initial contour.

In our approach, we match different templates to make the classification. For example, for human body detection, the object can be classified as a human based on successful head and limbs matching. To measure distance between the snake and templates, we use the chamfer distance map generated by the snake. Chamfer matching has been proved to be robust for object detection, but in the general setup, suitable threshold(s) for (e.g. the Canny) edge detection should be selected to avoid finding too many false negatives in the case of a cluttered/noisy scene. Moreover, the part of the target object, which corresponds to the pre-defined template, occurs in the image in a geometrically transformed way, and the template should be translated all over the image. Thus, generally, we have to search for the optimal parameters for an optimal affine transformation. Borgefors [3] introduced a hierarchical edge matching based on "coarse to fine" spatial resolution for this task. Huttenlocher et al. [6] suggested to search the transformation space using a cell decomposition strategy, see also [12], which idea became popular in applications. However, as it still needs quite a lot of computation/parallelization, the transformation space is usually reduced, like e.g. in [5] for the magnification parameter. Using the snake representation we can restrict this parameter space. The template is not translated blindly in the whole image, as we adjust its origin to snake points. Considering consequent snake points, we can estimate the direction of the snake, and thus can adjust the suitable rotation angle for the template. The magnification parameter can be bounded easily by adjusting it regarding the spatial size (e.g. diameter) of the snake. Moreover, we introduce a novel method to simplify the templates, which leads to better computational performance. We also observe how the density of the snake points and its simplification affect the reliability of matching.

The structure of the paper is as follows. In section 2 we shortly recall the chamfer matching method, and the GVF snake procedure. Section 3 describes how we can apply simplifications to make chamfer matching faster. In section 4, we present our experimental results.

## 2. PRELIMINARIES

### 2.1. Chamfer matching

Chamfer matching proposed by [1] is a matching procedure for binary images. While the Hamming distance can calcu-

late only the exactly matching pixel pairs, chamfer matching is more robust in case of less accurate matching. Let $S$ be a binary image, and $T = \{\mathbf{t}_i \in Z^2 \mid i = 1, \ldots, k; \; \mathbf{t}_1 = \mathbf{O}\}$ be a binary template, that is $T$ consists of $k$ ordered points starting at the origin. The distance map of $S$ is defined as $d_S : Z^2 \rightarrow R^+$, such that for an $\mathbf{x} \in Z^2$, $d_S(\mathbf{x})$ is the distance of $\mathbf{x}$ to the closest element of $S$. Trivially, $d_S(\mathbf{x}) = 0$ for all $\mathbf{x} \in S$. To create distance maps, one can use the Euclidean metric [4, 13], or the city-block/chessboard [11] or chamfer distances [2]. Moreover, we can find robust variants to weigh the distance values, like the truncated linear/quadratic [6, 14], or the exponentially decreasing one [10] to concentrate more on the regions closer to the edges. After obtaining $d_S$, a natural way to determine the goodness of fit between $S$ and $T$ at $\mathbf{x} \in Z^2$ is to consider the chamfer distance: $d_{\mathbf{x}}(S, T) = \frac{1}{k} \sum_{i=1}^{k} d_S(\mathbf{x} + \mathbf{t}_i)$.

Smaller $d_{\mathbf{x}}$ values correspond to better fit between $T$ and $S$ at $\mathbf{x}$. The median or the maximum of the distance values, the root mean square average, or to suppress the influence of outliers or missing data, the trimmed or truncated [6, 12] average can be also considered. In our experiments, without the loss of generality, we used the $\langle 3, 4 \rangle$ distance map [2].

## 2.2. Gradient Vector Flow (GVF) snakes

Snakes [8] start from an initial boundary, and move towards the actual boundary of the desired object through minimizing an energy functional. Internal forces prevent the snake from over-deforming, while external ones move the snake points towards the object boundary. The main idea behind GVF snakes [15] is to calculate a GVF force field based on the diffusion of the gradient values. The final snake is achieved by an iterative process, which can be tuned by weight parameters for the shape (elasticity, rigidity, viscosity) and the external field. We highlight only those elements which influence our matching process. In the case of a closed snake, its points are obtained in an ordered sequence $[\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_m = \mathbf{s}_1]$. The process for evolving the GVF snake considers the parameters $d_{\max}, d_{\min}$ for the maximum and minimum allowed distance between two snake points, respectively. In other words, if $D(\mathbf{s}_j, \mathbf{s}_{j+1 \mod m}) > d_{\max}$, then a new point is added between $\mathbf{s}_j$ and $\mathbf{s}_{j+1 \mod m}$, or if $D(\mathbf{s}_j, \mathbf{s}_{j+1 \mod m}) < d_{\min}$, either $\mathbf{s}_j$ or $\mathbf{s}_{j+1 \mod m}$ is removed, where $D$ is a typical distance function, and $1 \leq j \leq m$. It can be easily seen, that by requiring $d_{\max} < 1$, we can guarantee an 8-connected final snake. Small $d_{\max}$ values provides dense snakes for high computational cost. For this reason, an important point in our approach is to investigate how dense the snake can be for a reliable chamfer matching.

## 3. REDUCING THE NUMBER OF POINTS

We can adjust the density of the snake by its $d_{\max}$ parameter or we can reduce the number of points composing the template. We should remove points from the original template in such a way which is optimal according to the similarity measurement. So we should keep those points of the template that generate a distance map, which is "close" to the one generated by the original template. [16] had similar aims, but actual optimization was not given. Note that we also have to define the region within which the distance maps should be compared. Thus, we formulate the following problem:

**Problem.** *Reduce a binary set $A \subseteq B$ with $|A| = N$ to $K$ of its points, such that the distance map generated by the $K$ points is closest to the distance map generated by $A$ within $B$.*

The distance maps are compared by summing the differences between their corresponding values within $B$. Note that as we want to remove points from $A$, it is equivalent to obtain a distance map having minimal sum for its values within $B$. In finding an optimal way for the above reduction, we start with the case $A = B$. Consider the Voronoi decomposition of $B = \bigcup_{i=1}^{K} B_i$ according to some corresponding fixed $K$ generator points $\mathbf{c}_1, \ldots, \mathbf{c}_K$. It is obvious that to have the minimal sum of distance values inside the Voronoi cells $B_i$, $i = 1, \ldots, K$, the generator points should be the centroids of the corresponding cells, respectively. In other words, we minimize the within-class variance in such a way that also leads to a minimal sum of distance values within the whole $B$. This kind of distribution of points can be achieved using Centroidal Voronoi Tesselation (CVT) algorithms, see e.g. [7]. These iterative methods are initialized with a random selection of $K$ points within $B$. Then, in every iteration step, a Monte-Carlo sampling is executed to update the centroids of the cells.

Our task is not that simple here, as we also have to take care about $A \subset B$. Note that if $A$ and $B$ are close with respect to some Hausdorff set-metric, such that $\mathbf{c}_i \in B$ for all $i = 1, \ldots, K$ after every CVT iteration step, then we will have the same outcome as for the $A = B$ case. Thus, we propose the following modification of the CVT algorithm presented in [7]:

Step 0: Choose positive integer $q$ and constants $\alpha_1, \alpha_2, \beta_1, \beta_2$, such that $\alpha_2, \beta_2 > 0$, $\alpha_1 + \alpha_2 = \beta_1 + \beta_2 = 1$; choose an initial set of $K$ points $\mathbf{c}_1, \ldots, \mathbf{c}_K$ in $B$, e.g. by using a Monte Carlo method; set $j_i = 1$ for $i = 1, \ldots, K$;

Step 1: Choose $q$ points $\mathbf{y}_1, \ldots, \mathbf{y}_q$ in $B$ at random, e.g. by a Monte Carlo method, according to some probability density function (in our case uniformly);

Step 2: For $i = 1, \ldots, K$, gather together in the set $W_i$ all sampling points $\mathbf{y}_r$ closest to $\mathbf{c}_i$ among $\mathbf{c}_1, \ldots, \mathbf{c}_K$, i.e. lying in the Voronoi region of $\mathbf{c}_i$ w.r.t. $B$; if the set $W_i$ is empty, do nothing; otherwise, compute the $\mathbf{u}_i$ average of the set $W_i$ and set

$$\mathbf{c}_i \leftarrow \frac{(\alpha_1 j_i + \beta_1)\mathbf{c}_i + (\alpha_2 j_i + \beta_2)\mathbf{u}_i}{j_i + 1}, \quad j_i \leftarrow j_i + 1.$$

The new set of $\mathbf{c}_1, \ldots, \mathbf{c}_K$ along with the unchanged $c_j$'s (i.e. when $W_i$ is empty), form the new set of points $\mathbf{c}_1, \ldots, \mathbf{c}_K$;

Step 3: If for some $i$, $\mathbf{c}_i \notin A$ then $\mathbf{c}_i \leftarrow \mathbf{z}$, where $D(\mathbf{c}_i, \mathbf{z}) = \min_{\mathbf{y} \in A}\{D(\mathbf{c}_i, \mathbf{y})\}$, i.e. $\mathbf{z}$ is the nearest point to $\mathbf{c}_i$ in $A$ regarding some distance function $D$.

Step 4: Similarly to the CVT case, convergence can be shown. If the new points meet some convergence criterion, terminate; otherwise, return to Step 1.

For example, see Figure 1 to observe how the final positions of 16 centroids change according to the radius of the disc $A$ centered within the unit square $B$.
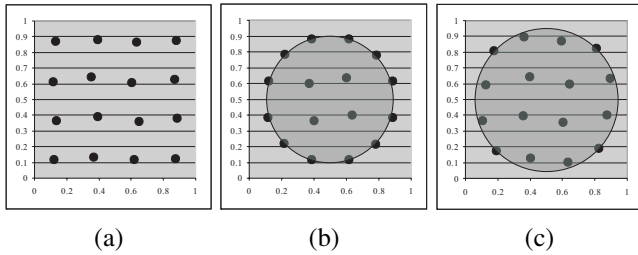


**Fig. 1**. The result of the modified CVT algorithm when $B$ is the unit square, and (a) $B = A$, (b) $A$ is a disc of radius 0.4, (c) $A$ is a disc of radius 0.45.

For a one-pixel wide digital template $A$, when the set $B$ is chosen to be close to it, a possible alternative is to keep equidistant points from $A$. See e.g. Figure 2 for a head template, where we applied different rates of reduction by keeping given percentages of template points.
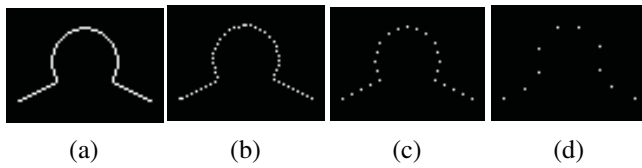


**Fig. 2**. Reduction of the number of template points for chamfer matching; (a) original template, (b) 50%, (c) 25%, (d) 10% of the template points.

We used the common $3 \times 3$ structuring element in dilations to obtain the regions (sets $B$) for checking the change of the distance map. Figure 3 depicts the result of some dilations.
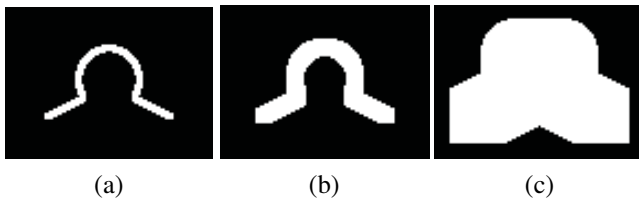


**Fig. 3**. Dilations of the head template to measure the change of the distance map within its region of support; (a) 1, (b) 3, (c) 12 dilations.

Figure 4 shows a quantitative comparison between the equidistant and CVT based reduction. Here, 25% of the template points were kept, and the calculation was made for several dilations (horizontal axis) of the original head template. The vertical axis shows the difference from the distance map generated by the original template. Our CVT approach is more important when larger regions around the target object are considered. The equidistant simplification is valid only for curve-like sets, while our proposed CVT algorithm can be applied for arbitrary sets, and also in the Euclidean space.
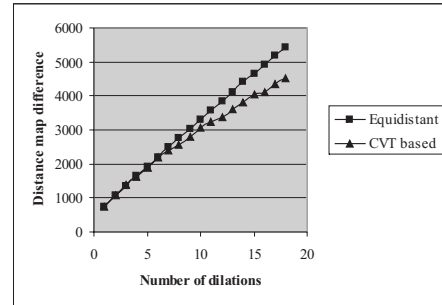


**Fig. 4**. The efficiency of the equidistant and CVT based reduction of one-pixel wide objects for chamfer matching.

Note that the relationship between the reduction rate and the computational performance is linear, which behaviour was also reflected in our tests. We also measured how the accuracy deteriorates with the simplification of the template. Generally, it does not depend on the selected $B$ region, and falls exponentially. Removing $80\% - 90\%$ of the points suggests larger risk for losing reliability of matching. The reduction rate can be adjusted using some test targets and the database templates. See Figure 5 for the corresponding results w.r.t. the head template. The horizontal axis shows the percentage of the template points kept, while the vertical one is the sum of the distance map values (accuracy) within the dilated regions.
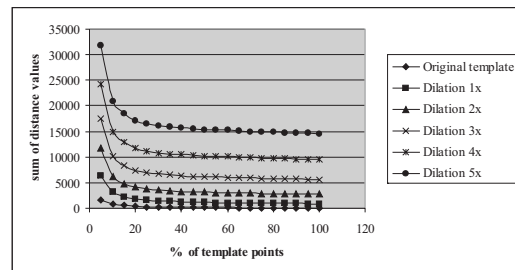


**Fig. 5**. The change of the distance map regarding the reduction of the head template at different levels.

## 4. MATCHING HUMAN BODY PARTS

In this section, we present some experimental results regarding matching human body parts, like the head and limbs. We also tested how the density of the snake and the reduction of the templates affected matching reliability. The best match is defined by the minimum of the sum of the corresponding distance map values. Figure 6 shows examples for matching results of the head and limb (leg) templates.



(a)                              (b)

**Fig. 6**. Best fitting positions of templates for a human silhouette represented by a snake; (a) head, (b) limb (leg).

We experienced that the simplification of the template and the consideration of a less dense snake can be a valid solution to speed up matching computations. Figure 7 shows the goodness-of-fit histograms for various densities of the snake and head template points. The template matcehd correctly in all these cases according to the minimal histogram value (normalized to 1). However, it is natural that the reliability of matching deteriorates, when less snake/template data are used. Similar results were found for the leg templates.
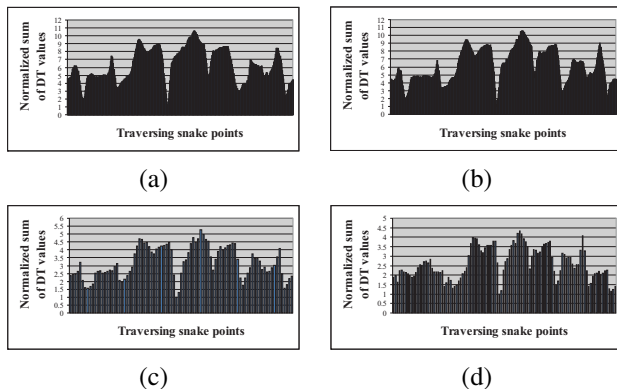


(a)                              (b)

(c)                              (d)

**Fig. 7**. Goodness of fit histograms for matching the head template; (a) 100% template points, $d_{\max} < 1$, (b) 25% template points, $d_{\max} < 1$, (c) 100% template points, $d_{\max} < 4$, (d) 25% template points, $d_{\max} < 4$.

## 5. REFERENCES

[1] H. G. Barrow, J. M. Tenenbaum, R. C. Bolles, and H. C. Wolf, "Parametric correspondence and chamfer matching: Two new techniques for image matching" *Int. Joint Conf. Artificial Intelligence*, Cambridge, MA, pp. 659-663, 1977.

[2] G. Borgefors, "Distance transformations in digital images" *Computer Vision, Graphics, and Image Processing*, Vol. 34(3), pp. 344-371, 1986.

[3] G. Borgefors, "Hierarchical chamfer matching: A parametric edge matching algorithm" *IEEE Trans. on Patt. Anal. Mach. Intel.*, Vol. 10(6), pp. 849-865, 1988.

[4] P.E. Danielsson, "Euclidean distance mapping" *Comput. Graph. Image Process.* Vol. 14, pp. 227-248, 1980.

[5] D. M. Gavrila and V. Philomin, "Real-time object detection for smart vehicles", *IEEE Int. Conf. on Comp. Vis.*, pp. 87-93, Kerkyra, Greece, 1999.

[6] D. Huttenlocher, G. Klanderman, and W. Rucklidge, "Comparing images using the Hausdorff distance" *IEEE Trans. on Patt. Anal. Mach. Intel.*, pp. 850-863, 1993.

[7] L. Ju, Q. Du and M. Gunzburger, "Probabilistic methods for centroidal Voronoi tessellations and their parallel implementations" *Parallel Computing* Vol. 28, pp. 1477-1500, 2002.

[8] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models", *Int. J. Comput. Vis.*, Vol. 1, pp. 321-331, 1987.

[9] M. Kumar, P.H.S. Torr and A. Zisserman, "Extending pictorial structures for object recognition", *British Machine Vision Conf.*, 2004.

[10] A. Pinz, M. Prantl, H. Ganster, "A robust affine matching algorithm using an exponentially decreasing distance function", *J. UCS*, Vol. 1(8), pp. 614-631, 1995.

[11] A. Rosenfeld and J. L. Pfaltz, "Distance functions on digital pictures", *Patt. Rec.* Vol. 1, pp. 33-61, 1968.

[12] W. Rucklidge, "Locating objects using the Hausdorff distance" *Int. Conf. on Comp. Vis.*, pp. 457-464, 1995.

[13] T. E. Schouten and E. L. van den Broek, "Fast exact euclidean distance (FEED) transformation", *ICPR 2004*, pp. 594-597, 2004.

[14] K. Toyama, A. Blake, "Probabilistic tracking with exemplars in a metric space", *IJCV*, Vol. 48, pp. 9-19, 2002.

[15] C.Y. Xu and J.L. Prince, "Snakes, Shapes, and Gradient Vector Flow", *IEEE Trans. on Im. Proc.*, Vol. 7(3), pp. 359-369, 1998.

[16] J. You, W. Zhu, E. Pissaloux and H. Cohen, "Hierarchical image matching: a chamfer matching algorithm using interesting points", *Real-time Imaging*, Vol. 1, pp. 245-259, 1995.